



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/016,982	12/14/2001	Vincent Auffray	FR920000065USI	4957

45092 7590 01/10/2007  
HOFFMAN, WARNICK & D'ALESSANDRO LLC  
75 STATE ST  
14TH FLOOR  
ALBANY, NY 12207

EXAMINER
----------

BASHORE, WILLIAM L

ART UNIT	PAPER NUMBER
----------	--------------

2176

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	01/10/2007	PAPER

**Please find below and/or attached an Office communication concerning this application or proceeding.**

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

<b>Office Action Summary</b>	Application No. 10/016,982	Applicant(s) AUFRAY ET AL.	
	Examiner William L. Bashore	Art Unit 2176	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

### Status

- 1) ☒ Responsive to communication(s) filed on 24 October 2006.
- 2a) ☐ This action is **FINAL**.                      2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

### Disposition of Claims

- 4) ☒ Claim(s) 1-12 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-12 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

### Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

### Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All    b) ☐ Some \*    c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

### Attachment(s)

- |  |   |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)  | 4) <input type="checkbox"/> Interview Summary (PTO-413)<br>Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)   | 5) <input type="checkbox"/> Notice of Informal Patent Application                       |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)<br>Paper No(s)/Mail Date <u>8/14/2007</u> . | 6) <input type="checkbox"/> Other: _____  |

### DETAILED ACTION

1. This action is responsive to communications: Request for Reconsideration (hereinafter the Request) filed 10/24/2006, to the original application filed 12/14/2001, with foreign priority filing date of 12/20/2000. IDS filed 10/24/2005, 2/6/2006, and 8/14/2006. Regarding submission on 8/14/2006, the references cannot be considered because copies of each reference cannot be found in the Record.
3. Claims 1-12 pending. Claim 1 is independent.

#### *Claim Objections*

4. Claims 4, 6-12 are objected to under 37 CFR 1.75(c) as being in improper form because multiple dependent claim 4 does not treat its dependencies in the singular. See MPEP § 608.01(n). Although said claims have been treated on the merits, nevertheless, correction is required.

#### *Claim Rejections - 35 USC § 103*

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

6. Claims 1-3, 9-12 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kikinis (hereinafter Kikinis), U.S. Patent No, 5,794,259 issued August 1, 1998, in view of Ashlin Weekly Update (hereinafter Ashlin Weekly), April 3, 1999, downloaded from <<http://www.ashlin.ca/news/HTM/19990403.htm>>, pages 1-2.

**In regard to independent claim 1**, Kikinis discloses filling in data on a displayed HTML form fetched from the Internet (Kikinin Abstract, column 2 lines 1-20, Figure 2).

Kikinis discloses control code in the form of a TSR program, or a plug-in module (typically downloaded) to a Web browser (a program component) (Kikinis column 3 lines 47-56).

Kikinis discloses that the plug-in is utilized for creation of bubble menus providing data to be filled in, said data pre-stored on a computer (typically contained in memory, i.e. hard drive, buffer memory, etc.). Data is then filled in the HTML form accordingly (Kikinis Figure 2, column 3 lines 35-36, 45-55, 59-63, column 4 lines 8-25). It is noted that Kikinis teaches a Web browser (i.e. Netscape) utilizing a plug-in for implementation of its invention (Kikinis column 3 lines 49-56). Buffers for holding data were known at the time of the invention, and Netscape uses a browser cache (a type of buffer) for holding specific information. Using a typical browser cache, Kikinis's browser can check for needed data stored (or pre-stored) in its own cache. If said data is not present, it will fetch the needed data from the relevant server. It is within reason that the skilled artisan can (if he/she notices that form fields remain empty) click the "Reload" button accordingly so as to fetch data from a server (another known browser feature). Accordingly, usage of a typical browser cache buffer within Kikinis fairly teaches Applicant's claimed limitation of determining whether data is already stored in the program buffer, and filling in said data if said data is present, while fetching said data from a server if the relevant data is not in the cache.

Kikinis does not specifically teach downloading said data from a server if said data is not already on the client, using said data for filling in said HTML form accordingly. However, Ashlin Weekly teaches Digitalme, comprising a proxy server which intercepts an Internet form, automatically filling in said form, and providing (downloading) the completed form for review (Ashlin Weekly page 1, third paragraph from top, compare with claim 1 "*upon determination (51) that said requested information data is not stored in the buffer memory allocated to the program component in the network client, the program component obtaining (57) the requested data by downloading the requested data from the network server and filling (59) the dedicated form fields in the hypertext document with the downloaded information data*"). It would have been obvious to one of ordinary

Art Unit: 2176

skill in the art at the time of the invention to apply Ashlin Weekly to Kikinis, allowing a user of Kikinis the capability of storing fill-in data (i.e. sensitive data) off of a client computer, and on a server instead, increasing the security and “privacy awareness” of a user’s information.

Kikinis teaches a typical form with empty fields displayed on a client browser (Kikinis Figure 2). As explained above, if needed data is not stored in the browser cache, Kikinis (via the well known use of a “Reload” button, implemented either automatically or manually), fetches data from a server. In this case, Kikinis would try to reload the entire Figure 2 form page. Secondary reference Ashlin Weekly’s proxy server intercepts Kikinis’s request, fetches the form page from the server, fills in the relevant data and returns the completed form. Since a browser Reload operation will replace (i.e. refresh) an old page with a new version of said page, Kikinis’s original Figure 2 page is now replaced with a new filled-in page. In this fashion, the needed data is downloaded from the proxy server, and the client accordingly fills in the fields on its own displayed blank copy by replacing the old page with the new (now filled-in) page, using Reload. The client has now completed its job of filling in the displayed form fields at the client.

**In regard to dependent claim 2**, Kikinis discloses a user perusing a form for accuracy, to which corrections can be made prior to uploading the completed HTML form to its destination (i.e. a server) (Kikinis column 2 lines 19-21, column 4 lines 5-9, also Figure 2 especially item 209).

**In regard to dependent claim 3**, Kikinis does not specifically teach user authentication prior to display of an HTML form document. However, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Kikinis to do this, since Kikinis teaches encryption and password protected access for user profiles (Kikinis column 4 lines 32-37), providing reasonable suggestion to the skilled artisan to extend user security for entering secure sites, providing the benefit of added security to sensitive Web sites (i.e. banking sites, etc.)

**In regard to dependent claim 9**, Kikinis discloses filling in forms on the Internet, said forms comprising Web forms (Kikinis column 3 lines 15-30, 32-33). It is well established that Web pages on the Internet utilize the HTTP protocol (i.e. <http://www.uspto.gov>, etc.).

**In regard to dependent claims 10-12**, claims 10-12 reflect the computer system, program, and computer program product comprising computer readable instructions used for performing the methods as claimed in claim 1, and are rejected along the same rationale.

7. **Claims 4-8 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kikinis and Ashlin Weekly as applied to claim 1 above, and further in view of Bhatia et al. (hereinafter Bhatia), U.S. Publication No. 2002/0154162 published October 24, 2002.**

**In regard to dependent claim 4**, Kikinis does not specifically teach scripting (script program). However, Bhatia teaches form fill in utilizing JavaScript (Bhatia paragraphs 0057, 0088). Bhatia teaches a JAVA Web server running scripts to capture data, to process captured data, or to present processed data (paragraph [0057]). Bhatia also teaches JavaScript in paragraph [0036] to represent code assistant objects, and paragraph [0068] teaches JavaScript for verifying a host environment and at least managing the Windows registry. Paragraph [0084] teaches JavaScript associated with Internet Explorer.

In addition, please note that Bhatia paragraph [0253] teaches using JavaScript to check if a user is logged in and the page from which the user is navigating from is an ecommerce form by scanning for specific keywords in the body text (i.e. address, state, etc.). Once said form is identified, element collection is executed (triggered) accordingly.

The above, combined with the well known use of JavaScript in data collection embodiments (i.e. a JavaScript username/password box, whereby control is managed, user input is collected in field(s), said input sent to a server for validation (or validated locally), etc.), renders obvious to the skilled artisan that JavaScript can be utilized for managing form contents, triggering download/upload of stored data, accordingly.

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Bhatia to Kikinis, providing Kikinis the benefit of JavaScript for increased platform independence.

**In regard to dependent claims 5-8,** Kikinis teaches categories of information (an identification list) (Kikinis Figure 2). Kikinis teaches a “bubble” of categories for types of data to be selected and filled in (Kikinis Figure 2 item 210). Since Bhatia teaches use of JavaScript for managing (see above), Bhatia’s JavaScript can be applied to manage Kikinis’s bubble selection.

Bhatia’s JavaScript can also be applied to send/receive data from a server accordingly (see above). Kikinis’s categories are a form of list. It would have been obvious to one of ordinary skill in the art to extend Kikinis’s bubble categorization list to a record list stored on a server, so that externally saved data can be itemized and categorized accordingly (categorization can be preserved), facilitating efficient retrieval of correct data.

Kikinis does not specifically teach frames. However, Bhatia teaches HTML forms with frames (Bhatia paragraph 0076). Bhatia teaches two frames in Figure 5 (at least a top shopping frame, and a bottom frame handling user login, etc.). Bhatia teaches a form filling service whereby each frame can contain a form (said forms can be managed using JavaScript) (Bhatia paragraph [0076]). Since both frames of Bhatia Figure 5 contain form input fields, and since both frames are related (the bottom frame controls cash back, and credit card information for the upper frame shopping site – see Bhatia paragraph [0063]), it would have been obvious to one of ordinary skill in the art at the time of the invention for Bhatia’s JavaScript to be referenced within each frame, so as to facilitate coordination of data, as well as for a more pleasing visual appearance.

Art Unit: 2176

Bhatia teaches form fill in utilizing JavaScript (i.e. applets, etc.) (Bhatia paragraphs 0057, 0088). It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Bhatia to Kikinis, providing Kikinis the benefit of JavaScript for increased platform independence.

### ***Response to Arguments***

8. Applicant's arguments filed 10/24/2006 have been fully and carefully considered but they are not persuasive.

Applicant argues on pages 6-7 of the Request that Ashlin does not teach the claimed limitations, because Ashlin's proxy does not download information from a server to fill in the intercepted forms. It is respectfully noted that the instant rejections have been clarified to indicate the use of browser cache and a "Reload" button. Browser caches reside on the client, and buffers for holding data were known at the time of the invention. Netscape uses a browser cache (a type of buffer) for holding specific information, such as Web pages. URL: <http://www.newhopepa.com/tekkorner/11-01-00.htm> (enclosed) describes how a typical browser cache works. Please note that this article addresses a cache as a "cache buffer" (end of paragraph 2).

As explained in the instant rejection, Ashlin intercepts a client page request (i.e. via Reload operation), fills in the form, and the Reload operation replaces the old page with Ashlin's new page at the client.

### ***Conclusion***

9. Any inquiry concerning this communication or earlier communications from the examiner should be directed to William L. Bashore whose telephone number is (571) 272-4088. The examiner can normally be reached on 11:30am - 8:00pm EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Heather Herndon can be reached on (571) 272-4136. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.



Art Unit: 2176

10. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

*William F. Bashore*  
**WILLIAM BASHORE**  
**PRIMARY EXAMINER**

January 7, 2007